

Ground Motion IM Procedure

This procedure outlines the process for downloading waveform and instrument data from GeoNet and using it to compute ground motion classifications. This is followed by computing ground motion intensity measures for the highest quality data.

Note: It is recommended that you set up the ground motion classifier and intensity measure modules in separate Python environments. This can easily be done with Conda. Be sure to install the latest version of the qcore modules, as many of the programs used in this procedure are dependent upon it.

Step-by-step guide

1. Download a list of earthquakes from GeoNet in CSV format: <https://quakesearch.geonet.org.nz> or with a query such as http://wfs.geonet.org.nz/geonet/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=geonet:quake_search_v1&outputFormat=csv&cql_filter=origintime>='2001-01-01'
 - a. Several lists can be appended to create a master file if downloaded in part using quake.
2. Use *mseed_write.py* to load station response information and download mseed data.
 - a. The data can be subset within the program to match specific years and/or magnitude ranges.
3. Run GMC (the ground motion classifier) on the downloaded data.
 - a. Create Konno matrices from gm_classifier directory. (example below):
 - i. `python gm_classifier/scripts/gen_konno_matrices.py dir_path/konno_matrices`
 - b. Extract features from gm_classifier directory. Because this can take a while, it is recommended to handle data by a yearly, or even monthly, basis. Records that have already been processed will not be reprocessed, so interrupted operations will not result in a massive loss of progress. (example below):
 - i. `python gm_classifier/gm_classifier/scripts/extract_features.py --ko_matrices_dir base_dir_path/konno_matrices base_dir_path/gmc_record/features/YEAR base_dir_path/mseed/YEAR`
 - c. Run model prediction (example below):
 - i. `python gm_classifier/scripts/predict.py base_dir_path/gmc_record/features/YEAR base_dir_path/gmc_record/YEAR/results.csv`
 - d. Concatenate results from all YEAR directories under *base_dir_path/gmc_record/* for a meta-results file.
4. Filter out the best results from the meta file with *gmc_filter.py*. This will copy files from the *mseed* directory to an *mseed_preferred* directory, while will be used for GM IM.
5. Pre-process the filtered results to produce 000, 090, and vertical outputs, which can be read in for ground motion intensity measures. This is most easily done with *process_observed.py* from the *Pre-processing* package, which can be request from the UC Earthquake Engineering group.
 - a. Note that this program can be run in parallel.
 - b. Currently, data are first demeaned, 5 seconds of zero padding are added to the beginning and end of the waveforms, instrument response is removed, the data is low-pass filtered to a corner frequency of 0.04 Hz, the data is rotated to ZNE components, and finally the zero-padded time windows are removed.
 - c. An example of this procedure is shown below:
 - i. `python geoNet/process_observed.py base_dir_path/mseed_preferred base_output_path text --n_procs 8`
6. Generate the ground motion intensity measure data. This requires the *IM_calculation* Python package. It is recommended to install this package in a new Python environment via Conda or otherwise. Run the program using a shell wrapper script. See the example, [im_runner.sh](#)
7. Finally, build the complete GM IM catalogue with by joining the ground motion classifier results with the ground motion intensity measures. An example for this is provided with the Python script *gm_im_merge.py*.



Related articles

- [Ground Motion IM Procedure](#)
- [Automated Earthquake Location Procedure](#)
- [Writing an automated test with Selenium](#)
- [Real-time Ground Motion Simulation Setup](#)
- [QuakeCoRE SW Team Code Review](#)